

# On converting any one-step method to a variational integrator of the same order

*George W. Patrick and Charles Cuell*  
*Applied Mathematics and Mathematical Physics*  
*Department of Mathematics and Statistics*

*Raymond J. Spiteri and William Zhang*  
*Department of Computer Science*

*University of Saskatchewan, Saskatoon, Saskatchewan, S7N 5E6 Canada*

*March 5, 2009*

## Abstract

In the formalism of constrained mechanics, such as that which underlies the SHAKE and RATTLE methods of molecular dynamics, we present an algorithm to convert any one-step integration method to a variational integrator of the same order. The one-step method is arbitrary, and the conversion can be automated, resulting in a powerful and flexible approach to the generation of novel variational integrators with arbitrary order.

## 1 Introduction

Consider a Lagrangian system defined by configuration space  $\hat{\mathcal{Q}} \equiv \{q\} \equiv \mathbb{R}^N$ , velocity phase space  $\mathbf{T}\hat{\mathcal{Q}} = \{(q, v)\} = \mathbb{R}^{2N}$ , and Lagrangian  $L: \hat{\mathcal{Q}} \rightarrow \mathbb{R}$ . Assume there is a holonomic constraint  $g(q) = 0$ ,  $g: \hat{\mathcal{Q}} \rightarrow \mathbb{R}^d$ , suppose  $g$  has full rank, and let  $\mathcal{Q} = g^{-1}(0)$ . The system evolves along curves  $q(t) \in \mathcal{Q}$  that are critical points of the action

$$S \equiv \int_a^b L(q(t), v(t)) dt \quad (1.1)$$

subject to the fixed endpoint constraints  $q(a), q(b)$  constant, and the first order constraint  $v(t) = q'(t)$ . This variational principle is equivalent to the Euler–Lagrange equations

$$\frac{dq^i}{dt} = v^i, \quad \frac{dv^i}{dt} = A^i(q, v), \quad (1.2)$$

where  $A(q, v)$  is found by solving the linear (Lagrange multiplier) problem (implicit sum on repeated indices)

$$\begin{aligned} \frac{\partial^2 L}{\partial v^i \partial v^j} A^j - \lambda_a \frac{\partial g^a}{\partial q^i} &= -\frac{\partial^2 L}{\partial v^i \partial q^j} v^j + \frac{\partial L}{\partial q^i}, \\ -\frac{\partial g^a}{\partial q^i} A^i &= \frac{\partial^2 g^a}{\partial q^i \partial q^j} v^i v^j. \end{aligned} \quad (1.3)$$

These are the general Lagrangian systems. For example, they specialize to the Euler equations for the motion of a rigid body [1, 12] by taking  $\hat{\mathcal{Q}}$  to be the  $3 \times 3$  matrices  $\{A\}$  and  $g^a$  to the the upper triangular

entries of  $AA^T - \mathbf{1}$ . And they specialize to the Kirchhoff approximations for the motion of an underwater vehicle [9, 15]. The same explicitly constrained formalism is exploited in the molecular dynamics algorithms SHAKE and RATTLE [2, 7, 8, 16].

The objective here is symplectic and momentum-preserving simulation of (1.2). Such simulations may be systematically generated by discretizing the defining variational principles, as in [11, 13, 17], and particularly [10], which is specific to the constrained formalism.

The theory of variational integrators is elaborated in [3, 4], based on geometric discretizations of the velocity phase space  $\mathbf{T}\mathcal{Q}$ , i.e., based on

1. certain assignments of *curve segments* to each  $(q, v) \in \mathbf{T}\hat{\mathcal{Q}}$ , where  $(q, v)$  is tangent to  $\mathcal{Q}$ ; and
2. a *discrete Lagrangian*  $L_h$  that approximates  $S_t$  for  $t = h$ ;  $S_t$  is defined to be the classical action (1.1) with  $a = 0$ ,  $b = t$  subject to (1.2), and satisfies

$$\frac{dS_t}{dt} = L(q^i, v^i). \quad (1.4)$$

The discrete variational principle is a finite-dimensional constrained optimization problem, in which the objective function is a sum of the discrete Lagrangian on sequences in  $\mathbf{T}\mathcal{Q}$ . If the curve segments associated with the elements of the sequence agree to order  $r$  with the exact evolution of the Lagrangian system and the discrete Lagrangian agrees to order  $r$  with the exact classical action, then the variational integrator is order  $r$  accurate [14].

By definition, any one-step numerical integration method of order  $r$  gives order  $r$  accurate solutions to Equations (1.2) and (1.4). Any such method can be used to provide the curve segments and discrete Lagrangian that are required to construct a variational integrator as outlined above. In this article we derive the variational integrator from the corresponding discrete Euler–Lagrange equations in terms of a one-step integrator of (1.2) and (1.4).

## 2 Basic Algorithm

Given a Lagrangian  $L$ , a constraint  $g$ , and a one-step numerical integrator of order  $r$ , which we call the *standard layer*, for the initial-value problem

$$\begin{aligned} \frac{dq^i}{dt} &= v^i, & \frac{dv^i}{dt} &= A^i(q, v), & \frac{dS_t}{dt} &= L(q, v), \\ q(0) &= q, & v(0) &= v, & S_t(0) &= 0, \end{aligned}$$

the aim is to generate a symplectic integrator of the same order. Let the standard layer be represented by

$$(q, v) \mapsto R_t(q, v) = (R_t^q(q, v), R_t^v(q, v), R_t^S(q, v)).$$

Assume that the standard layer exactly preserves the constraint; this restriction will be lifted later. By differentiating  $g(q) = 0$ , the space of vectors tangent to  $\mathcal{Q} = g^{-1}(0)$  is

$$\mathbf{T}\mathcal{Q} \equiv \{w \equiv (q, v) : \mathbf{D}g(q)v = 0\}.$$

Differentiating again,

$$\left. \frac{d}{d\epsilon} \right|_{\epsilon=0} \mathbf{D}g(q + \epsilon \delta q)(v + \epsilon \delta v) = v^T \mathbf{D}^2 g(q) \delta q + \mathbf{D}g(q) \delta v, \quad (2.1)$$

and so

$$\mathbf{T}\mathbf{T}\mathcal{Q} \equiv \{ \delta w \equiv ((q, v), (\delta q, \delta v)) : \mathbf{D}g(q) \delta q = 0, v^T \mathbf{D}^2 g(q) \delta q + \mathbf{D}g(q) \delta v = 0 \}.$$

The notation  $\mathbf{T}_w \mathbf{T}\mathcal{Q}$  denotes the vector space  $\{ \delta w : (w, \delta w) \in \mathbf{T}\mathbf{T}\mathcal{Q} \}$ .  $\mathbf{T}\mathcal{Q}$  and  $\mathbf{T}\mathbf{T}\mathcal{Q}$  are of course the first and second tangent bundle of the constraint. The notation for  $\mathbf{D}^2 g(q)$  is problematic; it is a bilinear form with values in  $\mathbb{R}^d$ . The quantity  $v^T \mathbf{D}^2 g(q)$ , where  $v \in \mathbb{R}^N$ , denotes the  $d \times N$  matrix  $v^i \partial^2 g^a / \partial q^i \partial q^j(q)$ .

We now construct the *symplectic layer* by defining the following quantities.

1. *The bias:* a pair of numbers

$$-1 \leq \alpha^- \leq 0, \quad 0 \leq \alpha^+ \leq 1,$$

such that  $\alpha^+ - \alpha^- = 1$ .

2. *The time step:* a number  $h > 0$ .

3. *The maps  $\partial_h^\pm$  and the discrete Lagrangian  $L_h$ :* the required curve segments are associated with each element  $(q, v)$  by  $t \mapsto R_t^q(q, v)$  and the discrete Lagrangian is obtained from  $(q, v) \mapsto R_h^S(q, v)$ .  $R_t^v$  is not used. The ends of the segments provide the maps

$$\partial_h^-(q, v) \equiv R_{h\alpha^-}^q(q, v), \quad \partial_h^+(q, v) \equiv R_{h\alpha^+}^q(q, v),$$

and the discrete Lagrangian

$$L_h(q, v) \equiv R_{h\alpha^+}^S(q, v) - R_{h\alpha^-}^S(q, v).$$

4. *The time step of the symplectic layer:* Given  $w_1 = (q_1, v_1) \in \mathbf{T}\mathcal{Q}$ , solve the following *discrete Euler–Lagrange equations* [4] for  $w_2 = (q_2, v_2) \in \mathbf{T}\mathcal{Q}$ :

$$\mathbf{D}L_h(w_1) \delta w_1 + \mathbf{D}L_h(w_2) \delta w_2 = 0,$$

$$\partial_h^+(w_1) = \partial_h^-(w_2),$$

for all  $\delta w_1, \delta w_2$  satisfying

$$\mathbf{D}\partial_h^-(w_1) \delta w_1 = 0, \quad \mathbf{D}\partial_h^+(w_2) \delta w_2 = 0,$$

$$\mathbf{D}\partial_h^+(w_1) \delta w_1 = \mathbf{D}\partial_h^-(w_2) \delta w_2,$$

$$(w_1, \delta w_1) \in \mathbf{T}\mathbf{T}\mathcal{Q}, \quad (w_2, \delta w_2) \in \mathbf{T}\mathbf{T}\mathcal{Q}.$$

It is not necessary that the *same* standard layer provides both  $\partial_h^-$  and  $\partial_h^+$ . For example, using any method to construct  $\partial_h^+$ , the *adjoint* [6] of the same method to construct  $\partial_h^-$ , and the bias  $\alpha^- = \alpha^+ = \frac{1}{2}$ , one evidently obtains a self-adjoint symplectic layer. Self-adjoint methods respect time reversal in the sense that a negative time step exactly reverses the discrete evolution. Also, self-adjoint methods are necessarily of even order: an odd-order self-adjoint method in fact has the next higher (even) order of accuracy because its odd order truncation errors must equal their negatives.

The symplectic layer corresponds to the finite-dimensional discrete variational principle of finding the critical points of the *discrete action*

$$S_h(w, \tilde{w}) \equiv L_h(w) + L_h(\tilde{w})$$

subject to the constraints

$$\begin{aligned}\partial_h^-(w) &= \text{constant}, & \partial_h^+(\tilde{w}) &= \text{constant}, \\ \partial_h^+(w) &= \partial_h^-(\tilde{w}), \\ g(q, v) &= 0, & g(\tilde{q}, \tilde{v}) &= 0,\end{aligned}$$

as discussed in [4].

By the general theory, the maps  $\partial_h^-$ ,  $\partial_h^+$  split  $\mathbf{T}\mathbf{T}\mathcal{Q} = \ker \mathbf{D}\partial_h^- \oplus \ker \mathbf{D}\partial_h^+$ ; i.e.,  $(w, \delta w)$  splits as

$$\begin{aligned}\delta w &= \delta w^+ + \delta w^-, \\ \delta w^+ &\in \ker \mathbf{D}\partial_h^-(w), & \delta w^- &\in \ker \mathbf{D}\partial_h^+(w).\end{aligned}$$

(In the second line, the presence of plus with minus is intentional and conforms to the notation of [4].) The *discrete Lagrange one-form* is defined by

$$\theta_{L_h}^-(w) \delta w \equiv -\mathbf{D}L_h(w) \delta w^-,$$

and the general theory assures that the symplectic layer is a symplectic integrator with respect to  $\omega_{L_h} \equiv -\mathbf{d}\theta_L^-$ .

In Lagrangian systems, the Noether theorem shows that the presence of continuous symmetry is equivalent to the presence of conserved momenta. For example, translational [rotational] symmetry implies conservation of linear [angular] momentum; see [1, 12] for the basic theory, some of which the discussion here must assume. The discrete Noether theorem provides the same symmetry-momentum equivalence for discrete Lagrangian systems: Suppose that a symmetry group  $\mathcal{G}$  acts on  $\hat{\mathcal{Q}}$ , such that

1.  $g$  is invariant;
2.  $\partial_h^+, \partial_h^-$  intertwine the lift of the action to  $\mathbf{T}\mathcal{Q}$ ; and
3.  $L_h$  is invariant.

Let  $\mathfrak{g}$  be the Lie algebra of  $\mathcal{G}$ , and let  $\xi \in \mathfrak{g}$ . Then the symplectic layer preserves the *discrete momentum* defined by

$$J_\xi(w) \equiv -\theta_{L_h}^-(w)\xi w,$$

where  $\xi \in \mathfrak{g}$  and  $\xi w$  is the infinitesimal generator of  $\xi$  at  $w$ . For example,  $\mathcal{G}$  could be a matrix Lie group that acts on  $\mathbb{R}^N$  by matrix multiplication, the exact Lagrangian  $L$  invariant, the constraint  $g$  invariant, and the standard layer an explicit Runge–Kutta method. Then the standard layer intertwines the action on  $\mathbf{T}\mathcal{Q}$ , and  $L_h$  is invariant, and the symplectic layer will preserve the corresponding discrete momenta.

Neither the discrete symplectic form nor the discrete momentum equals in general the continuous counterpart. For example, if the system is rotationally invariant then the symplectic layer need not preserve some familiar mechanical angular momentum. The discrete Lagrangian system has its own version of angular momentum, which is near to the continuous one, but through its dependence on  $\partial_h^-, \partial_h^+$ , is in general a complicated function of  $(q, v)$ .

### 3 Constrained algorithm

We require the following standard lemma, which will justify the use of a variety of (Lagrange) multipliers.

**Lemma 3.1.** *Suppose that  $A: \mathbb{E} \rightarrow \mathbb{F}$  and  $\mu: \mathbb{E} \rightarrow \mathbb{G}$  are linear. Let  $\mu$  be onto. Then  $A(e) = 0$  for all  $e$  such that  $\mu(e) = 0$  if and only if there is a  $\lambda: \mathbb{G} \rightarrow \mathbb{F}$  such that  $A = \lambda\mu$ .*

*Proof.* Suppose  $A$  is zero on  $\ker \mu$ . Then  $A$  drops to  $\bar{A}: \mathbb{E}/\ker \mu \rightarrow \mathbb{F}$ . Also,  $\mu$  drops to  $\bar{\mu}: \mathbb{E}/\ker \mu \rightarrow \mathbb{G}$ , and this is a linear isomorphism since  $\mu$  is onto. Set  $\lambda \equiv \bar{A}\bar{\mu}^{-1}$ . Then, letting  $\pi: \mathbb{E} \rightarrow \mathbb{E}/\ker \mu$  be the quotient map,  $\lambda\mu(e) = \bar{A}\bar{\mu}^{-1}\mu(e) = \bar{A}\pi(e) = A(e)$ . Conversely, if  $A = \lambda\mu$  and  $\mu(e) = 0$  then  $A(e) = \lambda(\mu(e)) = \lambda(0) = 0$ .  $\square$

We develop the equations used in the algorithm incrementally in stages; see Figures 1 and 2. In the figures, the dimension counts for the equations and variables are at right. The stages are equivalent representations of the same algorithm, starting from the fundamental description of the algorithm in Stage 0. Only the equations appearing in Stage 4 are implemented and solved in practice. In going from Stage 0 to Stage 4, we increase the number of equations to be solved from  $5(N - d)$  to  $12N + 5d$ .

*Stage 0: The fundamental algorithm, given directly on the constrained phase space, as in [4].* The fundamental algorithm is defined on  $\mathcal{Q} = g^{-1}(0)$ , regarded as a submanifold of  $\hat{\mathcal{Q}}$ . The constraints

$$\mathbf{D}\partial_h^-(w_1)\delta w_1 = 0, \quad \mathbf{D}\partial_h^+(w_2)\delta w_2 = 0,$$

are enforced with multipliers (in  $\mathbf{T}\mathcal{Q}$ )  $\lambda^-$  and  $\lambda^+$ , both having dimension  $\dim \mathcal{Q} = N - d$ . The constraint

$$\mathbf{D}\partial_h^+(w_1)\delta w_1 = \mathbf{D}\partial_h^-(w_2)\delta w_2,$$

is enforced with a multiplier  $\mu$ . Setting  $\delta w_1$  and  $\delta w_2$  alternately to zero gives the two equations (S0.2) and (S0.1), respectively. The connecting constraint  $\partial_h^+(w_1) = \partial_h^-(w_2)$  translates unchanged to (S0.3).

*Stage 1: Enforce the restriction  $\delta w_i \in \mathbf{T}_{w_i}\mathbf{T}\mathcal{Q}$  by introducing multipliers.* The restriction  $(\delta q, \delta v) = \delta w \in \mathbf{T}_w\mathbf{T}\mathcal{Q}$  is

$$\mathbf{D}g(q)\delta q = 0, \quad v^T \mathbf{D}^2g(q)\delta q + \mathbf{D}g(q)\delta v = 0.$$

For each equation in (S0.1) and (S0.2), there are two corresponding multipliers  $\nu_1, \nu_2$ , which are row vectors of length  $d$ .

*Stage 2: Disambiguate  $\lambda^-, \lambda^+, \mu$ .* The multipliers  $\lambda^+, \lambda^-, \mu$  are ambiguous as row vectors in  $\mathbb{R}^N$  up to any vector orthogonal to the constraint. Specifying them into  $\mathbf{T}\mathcal{Q}$  disambiguates them. Because  $\mathcal{Q} = g^{-1}(0)$  and the rows of  $\mathbf{D}g$  span the orthogonal complement to the tangent space of  $\mathcal{Q}$ , the multipliers should have zero dot product with the rows of  $\mathbf{D}g$ , e.g.,  $\lambda \mathbf{D}g(q)^T = 0$  for a multiplier  $\lambda$  at  $q \in \mathcal{Q}$ .

*Stage 3: Lift the restriction that the standard layer preserves the constraint.* A numerical integrator of order  $r$  used in the standard layer will in general preserve the constraint  $g$  only to accuracy order  $r$ . We posit a map  $\iota: \mathbb{R}^N \times \mathbb{R}^d \rightarrow \mathbb{R}^N$  such that

1.  $\iota(q, 0) = q$ ; and
2. if  $g(q) = 0$ , then  $\text{Im } \mathbf{D}\theta\iota(q, 0)$  is a complement of  $\ker \mathbf{D}g$ ; i.e.,  $\text{Im } \mathbf{D}\theta\iota(q, 0) \oplus \ker \mathbf{D}g(q) = \mathbb{R}^N$  for all  $q \in \mathcal{Q}$ .

<b>Stage 0:</b> Given $w_1 = (q_1, v_1) \in \mathbf{T}\mathcal{Q}$ solve, for $w_2 = (q_2, v_2) \in \mathbf{T}\mathcal{Q}$ , the equations	
(S0.1) $(\mathbf{D}L_h(w_1) = \lambda^- \mathbf{D}\partial_h^-(w_1) + \mu \mathbf{D}\partial_h^+(w_1))\delta w_1 \quad \forall \delta w_1 \in \mathbf{T}_{w_1}\mathbf{T}\mathcal{Q}$	$2(N-d)$
(S0.2) $(\mathbf{D}L_h(w_2) = \lambda^+ \mathbf{D}\partial_h^+(w_2) - \mu \mathbf{D}\partial_h^-(w_2))\delta w_2 \quad \forall \delta w_2 \in \mathbf{T}_{w_2}\mathbf{T}\mathcal{Q}$	$2(N-d)$
(S0.3) $\partial_h^+(w_1) = \partial_h^-(w_2)$	$\frac{N-d}{5N-5d}$
Lagrange multipliers $\lambda^-, \lambda^+, \mu$ time advanced state $(q_2, v_2) \in \mathbf{T}\mathcal{Q}$	$\frac{3(N-d)}{5N-5d}$
<b>Stage 1:</b>	
(S1.1) $\mathbf{D}L_h(w_1) = \lambda^- \mathbf{D}\partial_h^-(w_1) + \mu \mathbf{D}\partial_h^+(w_1) + \nu_1^- [\mathbf{D}g(q_1), 0] + \nu_2^- [v_1^T \mathbf{D}^2 g(q_1), \mathbf{D}g(q_1)]$	$2N$
(S1.2) $\mathbf{D}L_h(w_2) = \lambda^+ \mathbf{D}\partial_h^+(w_2) - \mu \mathbf{D}\partial_h^-(w_2) + \nu_1^+ [\mathbf{D}g(q_2), 0] + \nu_2^+ [v_2^T \mathbf{D}^2 g(q_2), \mathbf{D}g(q_2)]$	$2N$
(S1.3) $\partial_h^+(w_1) = \partial_h^-(w_2)$	$\frac{N-d}{5N-d}$
Lagrange multipliers $\lambda^-, \lambda^+, \mu$ Lagrange multipliers $\nu_1^+, \nu_2^+, \nu_1^-, \nu_2^-$ time advanced state $(q_2, v_2) \in \mathbf{T}\mathcal{Q}$	$\frac{3(N-d)}{5N-d}$
<b>Stage 2:</b>	
(S2.1) $\mathbf{D}L_h(w_1) = \lambda^- \mathbf{D}\partial_h^-(w_1) + \mu \mathbf{D}\partial_h^+(w_1) + \nu_1^- [\mathbf{D}g(q_1), 0] + \nu_2^- [v_1^T \mathbf{D}^2 g(q_1), \mathbf{D}g(q_1)]$	$2N$
(S2.2) $\mathbf{D}L_h(w_2) = \lambda^+ \mathbf{D}\partial_h^+(w_2) - \mu \mathbf{D}\partial_h^-(w_2) + \nu_1^+ [\mathbf{D}g(q_2), 0] + \nu_2^+ [v_2^T \mathbf{D}^2 g(q_2), \mathbf{D}g(q_2)]$	$2N$
(S2.3) $\partial_h^+(w_1) = \partial_h^-(w_2)$	$N-d$
(S2.4) $\lambda^- \mathbf{D}g(\partial_h^-(w_1))^T = 0, \quad \mu \mathbf{D}g(\bar{q})^T = 0, \quad \lambda^+ \mathbf{D}g(\partial_h^+(w_2))^T = 0$	$\frac{3d}{5N+2d}$
Lagrange multipliers $\lambda^-, \lambda^+, \mu$ Lagrange multipliers $\nu_1^+, \nu_2^+, \nu_1^-, \nu_2^-$ time advanced state $(q_2, v_2) \in \mathbf{T}\mathcal{Q}$	$\frac{3N}{5N+2d}$
<b>Stage 3:</b>	
(S3.1) $\mathbf{D}L_h(w_1) = \hat{\lambda}^- \mathbf{D}\hat{\partial}_h^-(w_1) + \hat{\mu}_1 \mathbf{D}\hat{\partial}_h^+(w_1) + \nu_1^- [\mathbf{D}g(q_1), 0] + \nu_2^- [v_1^T \mathbf{D}^2 g(q_1), \mathbf{D}g(q_1)]$	$2N$
(S3.2) $\mathbf{D}L_h(w_2) = \hat{\lambda}^+ \mathbf{D}\hat{\partial}_h^+(w_2) - \hat{\mu}_2 \mathbf{D}\hat{\partial}_h^-(w_2) + \nu_1^+ [\mathbf{D}g(q_2), 0] + \nu_2^+ [v_2^T \mathbf{D}^2 g(q_2), \mathbf{D}g(q_2)]$	$2N$
(S3.3) $(\hat{\lambda}^-)^T = \mathbf{D}\mathbb{P}(\hat{\partial}_h^-(w_1))^T (\lambda^-)^T, \quad (\hat{\lambda}^+)^T = \mathbf{D}\mathbb{P}(\hat{\partial}_h^+(w_2))^T (\lambda^+)^T$	$2N$
(S3.4) $\hat{\mu}_1^T = \mathbf{D}\mathbb{P}(\hat{\partial}_h^+(w_1))^T \mu, \quad \hat{\mu}_2^T = \mathbf{D}\mathbb{P}(\hat{\partial}_h^-(w_2))^T \mu$	$2N$
(S3.5) $\bar{q} = \mathbb{P}\hat{\partial}_h^+(w_1), \quad \hat{\partial}_h^-(w_2) = \iota(\bar{q}, \theta^+)$	$2N$
(S3.6) $q_1^- = \mathbb{P}\hat{\partial}_h^-(w_1), \quad q_2^+ = \mathbb{P}\hat{\partial}_h^+(w_2)$	$2N$
(S3.7) $g(q_2) = 0, \quad \mathbf{D}g(q_2)v_2 = 0$	$2d$
(S3.8) $\lambda^- \mathbf{D}g(q_1^-)^T = 0, \quad \mu \mathbf{D}g(\bar{q})^T = 0, \quad \lambda^+ \mathbf{D}g(q_2^+)^T = 0$	$\frac{3d}{12N+5d}$
Lagrange multipliers $\lambda^-, \lambda^+, \mu, \hat{\lambda}^+, \hat{\lambda}^-, \hat{\mu}_1, \hat{\mu}_2$ Lagrange multipliers $\nu_1^+, \nu_2^+, \nu_1^-, \nu_2^-$ variables $q_1^-, \bar{q}, q_2^+$ variable $\theta^+$ time advanced state $(q_2, v_2) \in \mathbf{T}\mathcal{Q}$	$\frac{7N}{12N+5d}$

Figure 1: Stages 0–3.

**Stage 4:** Given  $w_1 = (q_1, v_1) \in T\mathcal{Q}$  solve, for  $w_2 = (q_2, v_2) \in T\mathcal{Q}$ , the equations

$$(S4.1) \quad q_1^- = \mathbb{P}\hat{\partial}_h^-(w_1) \quad N$$

$$(S4.2) \quad \bar{q} = \mathbb{P}\hat{\partial}_h^+(w_1) \quad N$$

$$(S4.3a) \quad \hat{\lambda}^- \mathbf{D}_q \hat{\partial}_h^-(w_1) + \nu_1^- \mathbf{D}g(q_1) + \hat{\mu}_1 \mathbf{D}_q \hat{\partial}_h^+(w_1) + \nu_2^- v_1^T \mathbf{D}^2 g(q_1) - \mathbf{D}_q L_h(w_1) = 0 \quad N$$

$$(S4.3b) \quad \lambda^- \mathbf{D}g(q_1^-)^T = 0 \quad d$$

$$(S4.3c) \quad (\hat{\lambda}^-)^T = \mathbf{D}\mathbb{P}(\hat{\partial}_h^-(w_1))^T (\lambda^-)^T \quad N$$

$$(S4.4a) \quad \hat{\mu}_1 \mathbf{D}_v \hat{\partial}_h^+(w_1) + \nu_2^- \mathbf{D}g(q_1) + \hat{\lambda}^- \mathbf{D}_v \hat{\partial}_h^-(w_1) - \mathbf{D}_v L_h(w_1) = 0 \quad N$$

$$(S4.4b) \quad \mu \mathbf{D}g(\bar{q})^T = 0 \quad d$$

$$(S4.4c) \quad \hat{\mu}_1^T = \mathbf{D}\mathbb{P}(\hat{\partial}_h^+(w_1))^T \mu \quad N$$

$$(S4.5a) \quad \hat{\lambda}^+ \mathbf{D}_q \hat{\partial}_h^+(w_2) + \nu_1^+ \mathbf{D}g(q_2) - \hat{\mu}_2 \mathbf{D}_q \hat{\partial}_h^-(w_2) + \nu_2^+ v_2^T \mathbf{D}^2 g(q_2) - \mathbf{D}_q L_h(w_2) = 0 \quad N$$

$$(S4.5b) \quad \lambda^+ \mathbf{D}g(q_2^+)^T = 0 \quad d$$

$$(S4.5c) \quad (\hat{\lambda}^+)^T = \mathbf{D}\mathbb{P}(\hat{\partial}_h^+(w_2))^T (\lambda^+)^T \quad N$$

$$(S4.6a) \quad \mathbf{D}_v L_h(w_2) - \nu_2^+ \mathbf{D}g(q_2) - \hat{\lambda}^+ \mathbf{D}_v \hat{\partial}_h^+(w_2) + \hat{\mu}_2 \mathbf{D}\hat{\partial}_h^-(w_2) = 0 \quad N$$

$$(S4.6b) \quad \mathbf{D}g(q_2)v_2 = 0 \quad d$$

$$(S4.6c) \quad \hat{\mu}_2^T = \mathbf{D}\mathbb{P}(\hat{\partial}_h^-(w_2))^T \mu \quad N$$

$$(S4.7a) \quad \hat{\partial}_h^-(w_2) = \iota(\bar{q}, \theta^+) \quad N$$

$$(S4.7b) \quad g(q_2) = 0, \quad d$$

$$(S4.8) \quad q_2^+ = \mathbb{P}\hat{\partial}_h^+(w_2) \quad \frac{N}{12N + 5d}$$

Lagrange multipliers  $\lambda^-, \lambda^+, \mu, \hat{\lambda}^+, \hat{\lambda}^-, \hat{\mu}_1, \hat{\mu}_2$   $7N$

Lagrange multipliers  $\nu_1^+, \nu_2^+, \nu_1^-, \nu_2^-$   $4d$

variables  $q_1^-, \bar{q}, q_2^+$   $3N$

variable  $\theta^+$   $2d$

time advanced state  $(q_2, v_2) \in T\mathcal{Q}$   $\frac{2N}{12N + 5d}$

$$(S4.3a') \quad (\lambda^- + \mu) + \nu_1^- G_0 + \nu_2^- v_1^T \mathbf{D}^2 g(q_1) - h \mathbf{D}_q L(q_1, v_1) = 0$$

$$(S4.3b') \quad \lambda^- G_0^T = 0$$

$$(S4.4a') \quad (h\alpha^- \lambda^- + h\alpha^+ \mu) + \nu_2^- G_0 - hv_1^T M_0 - ha_0 = 0$$

$$(S4.4b') \quad \mu G_0^T = 0$$

$$(S4.5a') \quad (\lambda^+ - \mu) + \nu_1^+ G_0 + \nu_2^- v_1^T \mathbf{D}^2 g(q_1) - h \mathbf{D}_q L(q_1, v_1) = 0$$

$$(S4.5b') \quad \lambda^+ G_0^T = 0$$

$$(S4.6a') \quad hv_2^T M_0 - \nu_2^+ G_0 + h\alpha_0 - \lambda^+ + \mu = 0$$

$$(S4.6b') \quad G_0 v_2 = 0$$

$$(S4.7a') \quad (q_2 - \bar{q}) - \mathbf{D}g(q_1)^T \bar{\theta}^+ + h\alpha^+ v_2 = 0$$

$$(S4.7b') \quad G_0 (q_2 - \bar{q}) = 0$$

Figure 2: Stage 4: the derivatives in stage 3 are split into partial derivatives with respect to  $q$  and  $v$ , and the equations are reorganized. the approximates are correspondingly the bottom primed equations.

Define a map  $\mathbb{P}$  by

$$\mathbb{P}(\hat{q}) \equiv q, \quad \hat{q} = \iota(q, \theta), \quad g(q) = 0; \quad (3.1)$$

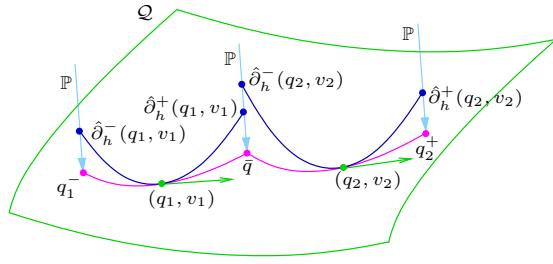
see the below figure. The map  $\mathbb{P}$  follows the constant  $\theta$ -fibers of  $\iota$  to where they intersect with  $\mathcal{Q}$ . The maps  $\partial_h^+, \partial_h^-$  are defined by the (constraint preserving) standard layer above. Letting the unconstrained standard layer define  $\hat{\partial}_h^+, \hat{\partial}_h^-$ , we redefine  $\partial_h^+, \partial_h^-$  by

$$\partial_h^+ \equiv \mathbb{P}\hat{\partial}_h^+, \quad \partial_h^- \equiv \mathbb{P}\hat{\partial}_h^-.$$

Also, we introduce the new variables

$$q_1^- \equiv \partial_h^-(w_1), \quad q_2^+ \equiv \partial_h^+(w_2), \\ \bar{q} \equiv \partial_h^+(w_1) = \partial_h^-(w_2),$$

with variable and equation count each of  $N$ .



Usually an explicit projection  $\mathbb{P}$  is not available and has to be computed iteratively. For example,  $\iota$  and  $\mathbb{P}$  may be naturally defined by

$$\begin{aligned} \mathbb{P}(\hat{q}) &= q, \\ \hat{q} &= q + \mathbf{D}g(q)^T \theta, \quad \theta \in \mathbb{R}^d, \\ g(q) &= 0. \end{aligned} \quad (3.2)$$

As  $\theta$  varies, this particular  $\iota(q, \theta)$  moves  $q \in \mathcal{Q}$  away from  $\mathcal{Q}$  orthogonally; the reverse, obtained from  $\mathbb{P}$ , projects  $\hat{q}$  orthogonally to  $\mathcal{Q}$ .

The connecting equation (S2.3) must be assumed to be full rank into  $\mathcal{Q}$ ; hence its (linearly independent) equation count is  $N - d$ . There are actually  $N$  equations when the image is considered into  $\mathbb{R}^N$ , as it must be for computations, but  $d$  of those are redundant because  $w_2 \in \mathcal{Q}$  by assumption and  $\partial_h^+, \partial_h^-$  preserve the constraints. In the second of (S3.5), explicitly writing the projection using new variable  $\theta^+$  resolves this problem because  $\theta^+$  robustly moves  $q$  away from the constraint  $\mathcal{Q}$ , i.e., locally linearly in a nondegenerate way. So Equation (S3.5) replaces (half of) Equation (S2.3), with equation count  $N$  rather than  $N - d$ , while the number of variables increases by  $d$  because that is the count for  $\theta^+$ .

*Stage 4: Split (S3.1) and (S3.2) into partial derivatives with respect to  $q$  and  $v$ ; rearrange terms and group equations.* The  $q$  and  $v$  partial derivatives of Equation (S3.1) give Equations (S4.3a) and (S4.4a), respectively. Similarly, partial derivatives of (S3.2) give (S4.5a) and (S4.6a). We group the equations so they can (eventually) be approximated by linear equations with the same coefficient matrices, as will be seen.

## 4 Implementation

We present here a strategy for the Stage 4 computation.

### 4.1 The vector field and its derivatives.

We now specialize to Lagrangians of the form

$$L \equiv \frac{1}{2}m_{ij}(q)v^i v^j + a_j(q)v^j - V(q). \quad (4.1)$$

This is the most general quadratic Lagrangian with configuration-dependent coefficients. As is easily verified, Equation (1.3) becomes

$$\begin{aligned} m_{ij}A^j - \lambda_a \frac{\partial g^a}{\partial q^i} &= -\Gamma_{ikl}v^k v^l - b_{ij}v^j - \frac{\partial V}{\partial q^i}, \\ -\frac{\partial g^a}{\partial q^i} A^i &= \frac{\partial^2 g^a}{\partial q^i \partial q^j} v^i v^j, \\ \Gamma_{ikl} &\equiv \frac{1}{2} \left( \frac{\partial m_{il}}{\partial q^k} + \frac{\partial m_{ik}}{\partial q^l} - \frac{\partial m_{kl}}{\partial q^i} \right), \\ b_{ij} &\equiv \frac{\partial a_i}{\partial q^j} - \frac{\partial a_j}{\partial q^i}. \end{aligned} \quad (4.2)$$

These are all linear equations for  $A$  and  $\lambda$  with coefficient matrix of the form

$$\begin{bmatrix} M(q) & -\mathbf{D}g(q)^T \\ -\mathbf{D}g(q) & 0 \end{bmatrix}, \quad M(q) \equiv [m_{ij}(q)]. \quad (4.3)$$

The algorithm requires the derivatives of the maps  $\partial_h^-$  and  $\partial_h^+$  as well as the derivative of  $L_h$ . Automatic differentiation [5] can be used to compute these by computing the derivative of the one-step method in the standard layer that defines them. Alternatively, by Lemma 4.1 of [6], the derivative of the standard layer  $R_t$  that uses a Runge–Kutta method may be computed as the same Runge–Kutta method applied to the equations of first variation. In this case, it is only required to determine the derivative of the vector field.

To compute the derivative of  $A^j$  with respect to  $q^m$  and  $v^m$ , we differentiate the first equation in (4.2):

$$\begin{aligned} m_{ij} \frac{\partial A^j}{\partial q^m} - \frac{\partial \lambda_a}{\partial q^m} \frac{\partial g^a}{\partial q^i} &= -\frac{\partial \Gamma_{ikl}}{\partial q^m} v^k v^l - \frac{\partial b_{ij}}{\partial q^m} v^j - \frac{\partial^2 V}{\partial q^i \partial q^m} - \frac{\partial m_{ij}}{\partial q^m} A^j - \frac{\partial^2 g^a}{\partial q^i \partial q^m} \lambda_a, \\ m_{ij} \frac{\partial A^j}{\partial v^m} - \frac{\partial \lambda_a}{\partial v^m} \frac{\partial g^a}{\partial q^i} &= -2\Gamma_{ikm} v^k \end{aligned}$$

and for the constraints

$$\begin{aligned} -\frac{\partial g^a}{\partial q^i} \frac{\partial A^i}{\partial q^m} &= \frac{\partial^3 g^a}{\partial q^i \partial q^j \partial q^m} v^i v^j \frac{\partial^2 g^a}{\partial q^i \partial q^m} A^i, \\ -\frac{\partial g^a}{\partial q^i} \frac{\partial A^i}{\partial v^m} &= 2 \frac{\partial^2 g^a}{\partial q^i \partial q^m} v^i. \end{aligned}$$

These resulting equations are linear in the required derivatives with coefficient matrix (4.3).

## 4.2 Fixed-point iteration

A possible approach for solving the Stage 4 (implicit) equations is by a fixed-point iteration. To find solutions to a generic equation  $f(x) = 0$ , split  $f = f_0 - \Delta f$  such that, for all  $b$ , an explicit solution to the equation  $f_0(x) = b$  is available. We call  $f_0$  an *approximate*. If  $\Delta f$  is sufficiently small, then in a suitable neighbourhood of the solution, the iteration  $x_{i+1} = f_0^{-1}(\Delta f(x_i))$  converges to a solution of  $f(x) = 0$ :

$$f_0(x) = \Delta f(x) = f_0(x) - f(x) \Leftrightarrow f(x) = 0.$$

The iteration is, given an initial iterate  $x_0$ ,

$$r_{i+1} = f_0(x_i) - f(x_i), \quad \text{solve } f_0(x_{i+1}) = r_{i+1},$$

or equivalently, after substituting  $f_0(x_i) = r_i$ ,

$$\begin{aligned} r_0 &= f_0(x_0), \\ r_{i+1} &= r_i - f(x_i), \\ \text{solve } f_0(x_{i+1}) &= r_{i+1}. \end{aligned} \tag{4.4}$$

This approach is useful for the Stage 4 equations because they are nonlinear and a good choice for  $f_0$  is generally available. In this way, the Stage 4 computation may be organized into its equations and corresponding approximates. The required solution is obtained by iteratively evaluating the equations themselves and then solving for the approximates using (4.4).

## 4.3 Stage 4 approximates

Because the time step is small, the various configurations  $q_1, \bar{q}, q_2$ , etc., are all close. Let  $G_0$ ,  $M_0$ , and  $a_0$ , be approximations to  $Dg(q)$ ,  $M(q)$ , and  $a(q)$  respectively, obtained by evaluation at some such configuration; e.g., the configuration  $q_1$  is a likely candidate.

Equations (S4.1), (S4.2), and (S4.8) are of the form of Equations (3.2), which, for sufficiently small  $h$ , can be effectively approximated by Taylor expansion of  $g(q) = 0$  at  $q = \hat{q}$ :

$$\begin{aligned} (\hat{q} - q) - G_0^T \theta &= 0, \\ -G_0(\hat{q} - q) - g(\hat{q}) &= 0. \end{aligned}$$

Equations (S4.3c), (S4.4c), (S4.5c), and (S4.6c) involve the derivative of  $\mathbb{P}$  in expressions such as

$$\hat{\lambda}^T = D\mathbb{P}(q)\lambda^T. \tag{4.5}$$

Differentiating equations (3.2) gives

$$\begin{aligned} \delta\hat{q} &= \delta q + Dg(q)^T \delta\theta + D^2(\theta^T g)(q) \delta q, \\ \delta g &= Dg(q) \delta q. \end{aligned} \tag{4.6}$$

The matrix  $D\mathbb{P}(q)$  is obtained by discarding  $\delta\theta$  after the inverse of (4.6), with  $\delta g = 0$ , i.e.,

$$D\mathbb{P}(q) = [\mathbf{1} \ 0] \begin{bmatrix} \mathbf{1} + D^2(\theta^T g)(q) & Dg(q)^T \\ Dg(q) & 0 \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{1} & 0 \\ 0 & 0 \end{bmatrix},$$

and Equation (4.5) becomes

$$\begin{bmatrix} \hat{\lambda}^T \\ 0 \end{bmatrix} = \begin{bmatrix} \mathbf{1} & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \mathbf{1} + \mathbf{D}^2(\theta^T g)(q) & \mathbf{D}g(q)^T \\ \mathbf{D}g(q) & 0 \end{bmatrix}^{-1} \begin{bmatrix} \lambda^T \\ 0 \end{bmatrix}.$$

If we define a variable  $z$  and put it in place of the zero in the matrix at left, then we can put a unit matrix in the (2,2) slot of the first matrix on the right side, and invert. The result is the linear equation

$$\begin{bmatrix} \mathbf{1} + \mathbf{D}^2(\theta^T g)(q) & \mathbf{D}g(q)^T \\ \mathbf{D}g(q) & 0 \end{bmatrix} \begin{bmatrix} \hat{\lambda}^T \\ z \end{bmatrix} = \begin{bmatrix} \lambda^T \\ 0 \end{bmatrix}. \quad (4.7)$$

This replaces (S4.3c), (S4.4c), (S4.5c), and (S4.6c), and one can use the approximate

$$\begin{bmatrix} \mathbf{1} & G_0^T \\ G_0 & 0 \end{bmatrix} \begin{bmatrix} \hat{\lambda}^T \\ z \end{bmatrix} = \begin{bmatrix} \lambda^T \\ 0 \end{bmatrix}, \quad (4.8)$$

which is computationally equivalent to

$$\begin{bmatrix} \mathbf{1} & -G_0^T \\ -G_0 & 0 \end{bmatrix} \begin{bmatrix} -\hat{\lambda}^T \\ z \end{bmatrix} = \begin{bmatrix} -\lambda^T \\ 0 \end{bmatrix}.$$

The remaining approximates are driven by the basic data of the variational principle:

$$\begin{aligned} \mathbf{D}_q L_h &\approx h \mathbf{D}_q L, \quad \mathbf{D}_v L_h \approx h v^T M_0 + h a_0, \\ \hat{\partial}_h^+(q, v) &\approx q + h \alpha^+ v, \quad \hat{\partial}_h^-(q, v) \approx q + h \alpha^- v. \end{aligned} \quad (4.9)$$

In order, the approximates for Equations (4.3a,b) obtained from (4.9),  $\hat{\lambda}^- \approx \lambda^-$ , and  $\hat{\mu} \approx \mu$  are as follows:

$$\begin{aligned} \mathbf{D}_q \partial_h^-(w_1) &\approx \mathbf{1}, \quad \hat{\lambda}^- \mathbf{D}_q \partial_h^-(w_1) \approx \lambda^-, \\ \mathbf{D}_q \partial_h^+(w_1) &\approx \mathbf{1}, \quad \hat{\mu}_1 \mathbf{D}_q \partial_h^+(w_1) \approx \mu, \\ \nu_1^- \mathbf{D}g(q_1) &\approx \nu_1^- G_0 = 0, \end{aligned}$$

resulting in

$$\begin{aligned} \lambda^- + \mu + \nu_1^- G_0 + \nu_2^- v_1^T \mathbf{D}^2 g(q_1) - h \mathbf{D}_q L(q_1, v_1) &= 0, \\ \lambda^- G_0^T &= 0. \end{aligned}$$

These are equations (S4.3a') and (S4.3b'). Similarly one obtains the approximates (S4.4a'b')–(S4.6a'b'), noting however that in (S4.5a') there are the further approximations

$$\nu_2^+ v_1^T \mathbf{D}^2 g(q_1) \approx \nu_2^- v_1^T \mathbf{D}^2 g(q_1), \quad \mathbf{D}_q L(q_2, v_2) \approx \mathbf{D}_q L(q_1, v_1).$$

Equations (S4.7a) and (S4.7b) are

$$\hat{\partial}_h^-(w_2) - \mathbf{D}g(\bar{q})^T \theta^+ = \bar{q}, \quad g(q_2) = 0,$$

which have to be solved for  $q_2$  and  $\theta^+$ . Taylor expanding the second equation at  $\bar{q}$ , and using  $g(\bar{q}) = 0$ , gives the approximate

$$\begin{aligned} q_2 - \bar{q} - \mathbf{D}g(q_1)^T \theta^+ + h \alpha^+ v_2 &= 0, \\ G_0(q_2 - \bar{q}) &= 0, \end{aligned}$$

which are Equations (S4.7a'b').

## 4.4 Stage 4 solution

The approximates (S4.3a'b') through (S4.5a'b') are solvable for the multipliers

$$\lambda^-, \mu, \lambda^+, \nu_1^-, \nu_2^-, \nu_1^+,$$

and the linear equations are all of the form (4.8). Indeed, one adds  $\alpha^-$  times (S4.3b') and  $\alpha^+$  times (S4.4b'), and then that together with (S4.4a') can be solved for  $\alpha^- \lambda^- + \alpha^+ \mu$  and  $\nu_2^-$ . Then, and similarly, (S4.3a'b') can be solved for  $\lambda^- + \mu$  and  $\nu_1^-$ . Together these give  $\lambda^-, \mu, \nu_1^-, \nu_2^-$ . Because  $\mu$  is then known, (S4.5a') with (S4.5b') minus (S4.4b') provide  $\lambda^+$  and  $\nu_1^+$ . In the same way (S4.6a'b') may be used to update  $\nu_2^+$  and  $v_2$ . Finally, (S4.7a'b') are solved for  $q_2$  and

$$\hat{\lambda}^-, \hat{\mu}_1, \hat{\mu}_2, \hat{\lambda}^+$$

occurring in (S4.3c), (S4.4c), (S4.5c), and (S4.6c'), may all be updated using appropriate versions of (4.7) and its approximates. The entire procedure can then be iterated until the variables  $q_2$  and  $v_2$  are at a predetermined accuracy.

It should be noted that the multipliers that impose the constraints  $g$ , i.e.,  $\nu_1^-, \nu_2^-, \nu_1^+, \nu_2^+$  are not unique. For example, doubling  $g$  results in halving these multipliers. Such multipliers are nonphysical, and convergence of the iteration of Stage 4 should not be bound to the convergence of the multipliers themselves. Rather, the degree of convergence can be determined from products such as  $\nu_2^- \mathbf{D}g(q_1)$ , which generally have the physical meaning of *force of constraint*; i.e., they are added directly in the equations to quantities with a physical interpretation.

## References

- [1] R. Abraham and J. E. Marsden. *Foundations of Mechanics*. Addison-Wesley, second edition, 1978.
- [2] H. C. Anderson. Rattle: A “velocity” version of the shake algorithm for molecular dynamics. *J. Comput. Phys.*, 52:24–34, 1983.
- [3] C. Cuell and G. W. Patrick. Skew critical problems. *Regul. Chaotic Dyn.*, 12:589–601, 2007.
- [4] C. Cuell and G. W. Patrick. Geometric discrete analogues of tangent bundles and constrained Lagrangian systems, 2008. arXiv:0807.1511v1 [math-ph].
- [5] A. Griewank. *Evaluating derivatives. Principles and techniques of algorithmic differentiation*. SIAM, 2000.
- [6] E. Hairer, C. Lubich, and G. Wanner. *Geometric numerical integration. Structure-preserving algorithms for ordinary differential equations*. Springer-Verlag, 2006.
- [7] B. Leimkuhler and S. Reich. *Simulating Hamiltonian dynamics*. Cambridge University Press, 2004.
- [8] B. J. Leimkuhler and R. D. Skeel. Symplectic numerical integrators in constrained Hamiltonian systems. *J. Comput. Phys.*, 112:117–125, 1994.
- [9] N. E. Leonard and J. E. Marsden. Stability and drift of underwater vehicle dynamics: mechanical systems with rigid motion symmetry. *Physica D*, 105:130–162, 1997.
- [10] S. Leyendecker, J. E. Marsden, and M. Ortiz. Variational integrators for constrained dynamical systems. *Z. Angew. Math. Mech. (ZAMP)*, 88:677–708, 2008.
- [11] J. E. Marsden, G. W. Patrick, and S. Shkoller. Multisymplectic geometry, variational integrators, and nonlinear PDEs. *Comm. Math. Phys.*, 199:351–395, 1998.
- [12] J. E. Marsden and T. S. Ratiu. *Introduction to Mechanics and Symmetry*, volume 17 of *Texts in Applied Mathematics*. Springer-Verlag, 1994.
- [13] J. E. Marsden and M. West. Discrete mechanics and variational integrators. *Acta Numerica*, 10:357–514, 2001.
- [14] G. W. Patrick and C. Cuell. Error analysis of variational integrators of unconstrained lagrangian systems. arXiv:0807.1516v1 [math.NA].
- [15] G. W. Patrick, R. M. Roberts, and C. Wulff. Stability transitions for axisymmetric relative equilibria of Euclidean symmetric Hamiltonian systems. *Nonlinearity*, 21:325–352, 2008.
- [16] J.-P. Ryckaert, G. Cicotti, and H. J. C. Berendsen. Numerical integration of the cartesian equations of motion of a system with constraints: molecular dynamics of n-alkanes. *J. Comput. Phys.*, 23:327–341, 1977.

[17] J. M. Wendlandt and J. E. Marsden. Discrete integrators derived from a discrete variational principle. *Physica D*, 106:233–246, 1997.